D212 Performance Assessment - Dimensionality Reduction Methods

Name: Coots, Anthony.
Affiliation: Grad Student M.Sc Data Analytics.
Date: `2024-05-30`
Version: 1.1.0, r1.

Introduction

"In this task, you will act as an analyst and create a data mining report. You must select one of the data dictionary and data set files to use for your report from the following web link: "Data Sets and Associated Data Dictionaries.""

- WGU

Competencies

4030.6.5 : Dimension Reduction Methods

- The graduate implements dimension reduction methods to identify significant variables.

Scenario

"One of the most critical factors in patient relationship management that directly affects a hospital's long-term cost-effectiveness is understanding the patients and the conditions leading to hospital admissions. When a hospital understands its patients' characteristics, it is better able to target treatment to patients, resulting in a more effective cost of care for the hospital in the long term.

You are an analyst for a hospital that wants to better understand the characteristics of its patients. You have been asked to use PCA to analyze patient data to identify the principal variables of your patients, ultimately enabling better business and strategic decision-making for the hospital."

- WGU

Table of Contents:

Research Question

A1: Proposal of Question

Propose one question that can be answered using PCA and is relevant to a real-world organizational situation.

Principal Component Analysis (PCA) is commonly used to transform a data set with continuous variables into a new set of variables known as principal components. These principal components aim to retain the maximum amount of variance in the data while performing dimensionality reduction of the original data set. The '*medical_clean*' data set which is provided in comma separated value format, contains thirteen different continuous variables. These thirteen can be reduced to fewer principal components, while perserving the variance within the data. Since the scenario at hand pertains to a hospital setting, the following question is proposed:

Question: "*What are the key patterns in the data set, and how can PCA be utilized to preserve and interpret the variation among the continuous variables to improve healthcare efficiency and efficacy*?"

A2: Defined Goal

Define one reasonable goal of the data analysis that is within the scope of the scenario and is represented in the available data.

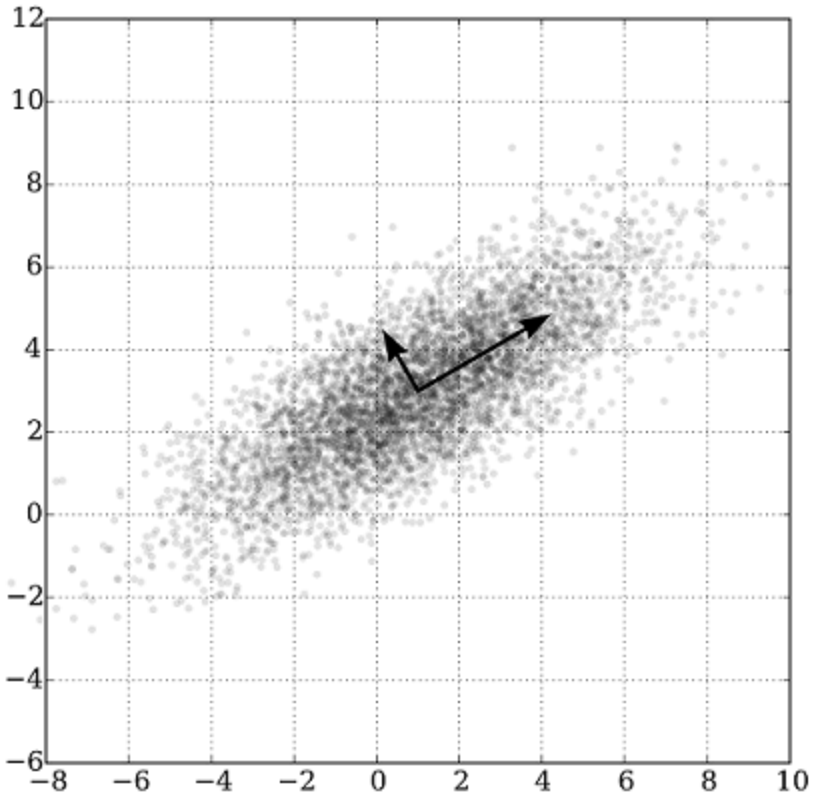
One goal of utilizing PCA to the '*medical_clean*' data set is to perform dimensionality reduction by taking the thirteen available continuous variables and forming them into fewer principal components. This aims to retain the maximum amount of variance in the data in order to facilitate a more efficient analysis. By doing so, the principal components that are derived from the continuous variables will give weighted scores that can help identify (and interpret) the key patterns in the data. These patterns may also help healthcare professionals identify patterns/factors that influence the efficiencies and efficacy of the hospital itself.

Method Justification

B1: Explanation of PCA

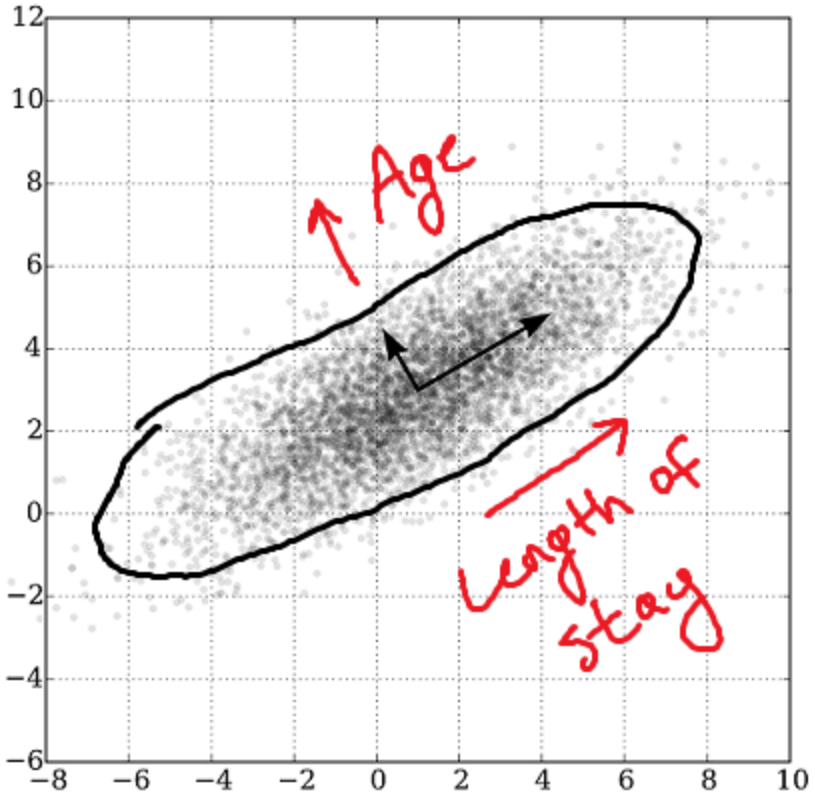
Logically explain how PCA analyzes the selected data set and include expected outcomes.

PCA starts by calculating a covariance matrix of the data measuring the covariance between pairs of variables. The matrix is then decomposed into eigenvectors and eigenvalues. Each vector represents a principal component direction of the feature space and each value indicates how much variance the component captures. The scatterplot below is an example illustration of how PCA analyzes the data set.



By calculating a covariance matrix of the variables and then determining the eigenvalues and eigenvectors. Each eigenvector represents a principal component where the corresponding eigenvalue is the amount of variance that the principal component captures from the data. For example, if variables such as age and length of hospital stay are correlated, PCA may combine these two variables together into a single principal component that summarizes both variables, reducing the overall dimensionality of the data set given by both variables alone. The expected outcomes of utilizing PCA include a reduced number of dimensions that simplify the data set while preserving as much of the variance in the data as possible. This reduction may provide insight into patterns in the

data, where each principal component may reveal different factors in hospital operation.



Note: This is a concept illustration. Example image provided by (Principal component analysis 2024.)

B2: PCA Assumption

Summarize one assumption of PCA.

An assumption of PCA is the linearity of the given data set, in this analysis, '*medical_clean*.' This suggests that the data has linear combinations of the variables. In the '*medical_clean*' data set, one could assume the total daily charge and number of days a patient has spent in the hospital variables likely have a linear relationship. This would then imply that the daily cost of hospitalization likely increases with the length of stay as a reflection of lengthy care, which is not entirely an irrational prediction of healthcare. Considering the linearity assumption, PCA would then likely map these correlated variables as a principal component that would represent both variables, reducing the dimensionality and keeping important information. Side note, as correlation is mentioned, it is equally important to recognize a key fact of data analytics, that correlation does not imply causality.

Data Preparation

C1: Continuous Data Set Variables

Accurately identify the continuous data set variables needed to answer the PCA question.

In order to identify the continuous data set variables needed to answer the PCA question, it is best to find all of the continuous data set variables available in the data set. The following is a list of continuous variables in the data set as identified in the provided data set, '*medical_clean*.'

- *Lat*
- *Lng*
- *Population*
- *Children*
- *Age*
- *Income*
- *VitD_levels*
- *Doc_visits*
- *Full_meals_eaten*
- *vitD_supp*
- *Initial_days*
- *TotalCharge*
- *Additional_charges*

In the '*medical_clean*' data set, variables like '*Population*' and '*Children*' are classified as discrete, opposed to continuous as they are counts. However, for PCA they are treated as continuous data points. Doing so is justified because of the range/distribution/contributing variance when under continuous variable treatment. PCA benefits from a scaled data set to ensure consistency and avoiding disproportionate influence from a variable much larger than '*Children*' like '*TotalCharge*' which is in the thousands range.

C2: Standardization of Data Set Variables

Standardize the continuous data set variables and include a cleaned data set.

In [2]:
```
import matplotlib.pyplot as plt
import numpy as np
import os
import pandas as pd
from sklearn.decomposition import PCA
from sklearn.preprocessing import StandardScaler
```

In [3]:
```
# What is my current working directory?
print("\n\n Current Working Directory: " + os.getcwd() + '\n')
```

```
 Current Working Directory: C:\Users\acoots\Desktop\Personal\Education\WGU\Data Ana
lytics, M.S\D212 - Data Mining II\Task 2 - Dimensionality Reduction Methods
```

In [4]:
```
# Read data into DataFrame.
df = pd.read_csv("medical_clean.csv")
```

In [5]:
```
continuous_data_set_var = [
    'Lat', 'Lng', 'Population', 'Children',
    'Age', 'Income', 'VitD_levels', 'Doc_visits',
    'Full_meals_eaten', 'vitD_supp', 'Initial_days',
    'TotalCharge', 'Additional_charges'
]

df_continuous = df[continuous_data_set_var].copy()
```

In [6]:
```
scaler = StandardScaler()
df_scaled = scaler.fit_transform(df_continuous)
df_scaled = pd.DataFrame(df_scaled, columns = df_continuous.columns)
```

In [7]:
```
df_scaled.head(5)
```

Out[7]:

	Lat	Lng	Population	Children	Age	Income	VitD_levels	Doc_visit
0	-0.814668	0.297134	-0.473168	-0.507129	-0.024795	1.615914	0.583603	0.94464
1	-1.463305	0.395522	0.090242	0.417277	-0.121706	0.221443	0.483901	-0.96798
2	0.886966	-0.354788	0.482983	0.417277	-0.024795	-0.915870	0.046227	-0.96798
3	0.952530	-0.149403	-0.526393	-0.969332	1.186592	-0.026263	-0.687811	-0.96798
4	-0.213252	0.943984	-0.315586	-0.507129	-1.526914	-1.377325	-0.260366	-0.01166

In [8]:
```
# Export cleaned dataset to csv.
df_scaled.to_csv("analysis_ready_medical_clean.csv", index = False)
```

Analysis

D1: Principal Components

Determine the matrix of all the principal components.

In [14]:
```
# Reusable code from D206!

# Quantitative (continous) variables only.
pca_cols = df_scaled[[
    'Lat', 'Lng', 'Population', 'Children',
    'Age', 'Income', 'VitD_levels', 'Doc_visits',
    'Full_meals_eaten', 'vitD_supp', 'Initial_days',
    'TotalCharge', 'Additional_charges'
]]

# Normalization to ensure PCA algorithm captures appropriate variance of data.
cols_normalized = (pca_cols - pca_cols.mean())/pca_cols.std()

# Sets the number of components, 13 in this case.
pca = PCA(n_components=pca_cols.shape[1])

# Used for dimension reduction, calculating the eigenvalues and eigenvectors.
pca.fit(cols_normalized)

# Stores the analysis into a local data frame.
med_pca = pd.DataFrame(pca.transform(cols_normalized), columns = [
    'PCA1', 'PCA2', 'PCA3', 'PCA4', 'PCA5', 'PCA6', 'PCA7', 'PCA8',
    'PCA9', 'PCA10', 'PCA11', 'PCA12', 'PCA13'])
# Organizes columns and respective pca values.
loadings = pd.DataFrame(pca.components_.T,
                        columns = ['PCA1', 'PCA2', 'PCA3', 'PCA4',
                                   'PCA5', 'PCA6', 'PCA7', 'PCA8',
                                   'PCA9', 'PCA10', 'PCA11', 'PCA12',
                                   'PCA13'],
                        index=pca_cols.columns)

print(loadings)
```

	PCA1	PCA2	PCA3	PCA4	PCA5
Lat	-0.018834	0.000913	-0.715570	-0.036559	0.128188
Lng	-0.011011	0.009716	0.274895	-0.474659	-0.554592
Population	0.028719	-0.029027	0.626046	0.295638	0.250669
Children	0.034537	0.017244	-0.034510	0.344621	0.158969
Age	0.084650	0.700793	0.011244	-0.020860	0.010691
Income	-0.019701	-0.019176	0.075776	-0.067301	0.412381
VitD_levels	-0.001995	0.020340	-0.020176	0.526197	-0.213021
Doc_visits	-0.006991	0.015446	0.017291	0.096735	0.282211
Full_meals_eaten	-0.020712	0.031960	-0.103248	0.454738	-0.385982
vitD_supp	0.025381	0.014511	0.029741	-0.262904	0.377611
Initial_days	0.699994	-0.089859	-0.022902	-0.007101	-0.018751
TotalCharge	0.701146	-0.079267	-0.020888	-0.003830	-0.019601
Additional_charges	0.085029	0.700745	0.013730	-0.004630	0.019713

	PCA6	PCA7	PCA8	PCA9	PCA10
Lat	-0.018260	-0.039974	-0.005117	-0.067661	-0.039423
Lng	-0.289613	0.229759	0.320779	0.056053	0.033702
Population	0.142253	-0.174676	-0.135732	-0.083567	0.038751
Children	0.231131	0.427505	0.717166	-0.131085	0.292473
Age	0.011755	0.006632	-0.017856	-0.013308	-0.020631
Income	-0.149024	0.651545	-0.162893	0.461862	-0.359436
VitD_levels	-0.366372	-0.208667	0.305325	0.061710	-0.634109
Doc_visits	-0.820104	0.040698	-0.076493	-0.285582	0.381544
Full_meals_eaten	-0.050904	0.062235	-0.238447	0.590939	0.462602
vitD_supp	-0.097049	-0.508283	0.424062	0.565530	0.137073
Initial_days	-0.017957	0.013322	-0.023812	0.008988	-0.007002
TotalCharge	-0.019199	0.012132	-0.022769	0.009702	-0.005149
Additional_charges	0.016979	0.006236	-0.025023	-0.006886	-0.010633

	PCA11	PCA12	PCA13
Lat	0.679459	0.008903	0.001359
Lng	0.384029	-0.004863	-0.000429
Population	0.615001	0.016751	-0.000658
Children	-0.006222	0.003440	-0.000938
Age	-0.001154	0.706577	0.026277
Income	0.056064	0.002441	0.001318
VitD_levels	-0.003265	-0.002389	-0.001497
Doc_visits	-0.056573	0.000868	-0.001114
Full_meals_eaten	0.073982	0.010748	-0.001632
vitD_supp	-0.018434	0.000367	-0.000604
Initial_days	0.000171	0.031504	-0.706271
TotalCharge	0.000729	-0.031475	0.706491
Additional_charges	0.020560	-0.705862	-0.036739

D2: Identification of the Total Number of Components

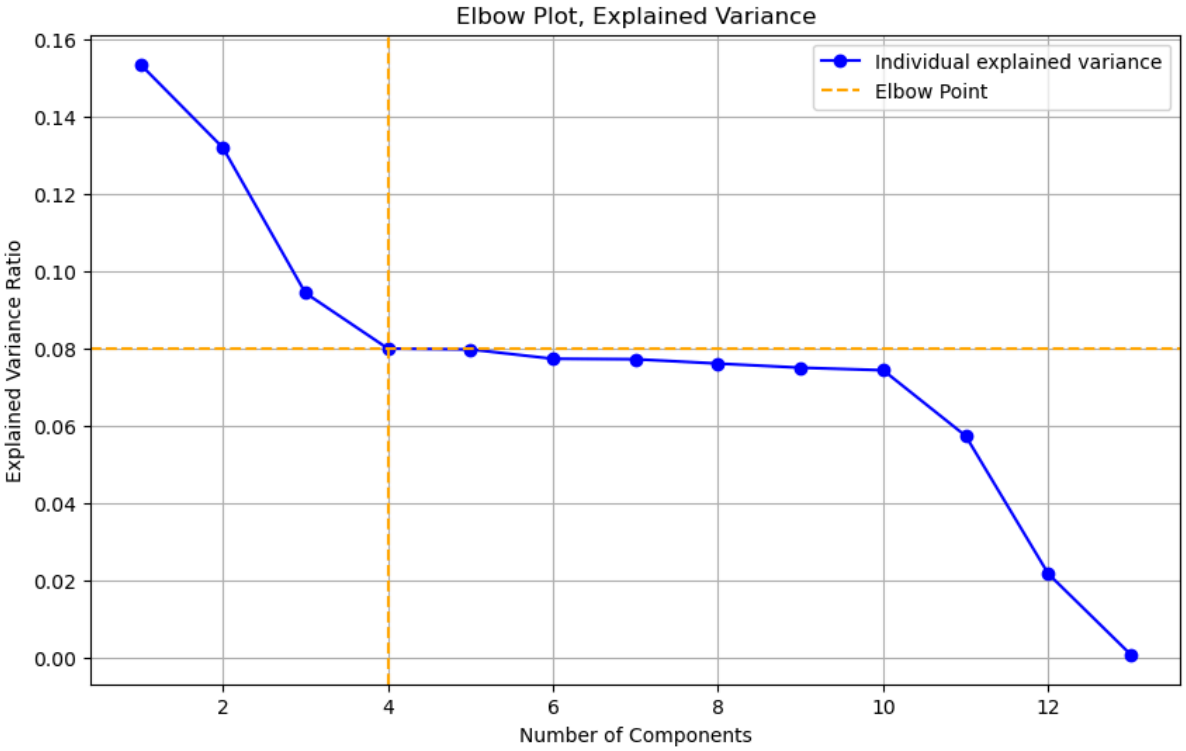
Identify the total number of principal components using the elbow rule.

In [15]:
```
explained_vari = pca.explained_variance_ratio_
cumulative_vari = np.cumsum(explained_vari)

# Display/plot size.
plt.figure(figsize=(10, 6))
# Plot for range of count of principal components.
plt.plot(range(1, len(explained_vari) + 1), explained_vari, marker='o', linestyle=

# Display touch-up.
plt.title('Elbow Plot, Explained Variance')
plt.xlabel('Number of Components')
plt.ylabel('Explained Variance Ratio')
plt.grid(True)
plt.axvline(x = 4, color = 'Orange', linestyle = '--', label = 'Elbow Point')
plt.axhline(y = 0.08, color = 'Orange', linestyle = '--')
plt.legend(loc='best')

# Show plot.
plt.show()
```



For this analysis, the total number of principal components is determined using an elbow rule (plot seen above). The curve starts to flatten after the fourth component, indicating that the additional variance that is explained by more components beyond the fourth is small. The spot selected is a good balance between keeping a low number of dimensions while retaining the maximum amount of variance in the data.

D3: Variance of Each Component

Identify the variance of each of the four principal components.

The first principal component explains about 15% of the variance in the data set from a pattern between '*Initial_days*' and '*TotalCharge*', with loadings of approximately 0.7 each. This highlights the time spent and financial dimentions of patient admissions in the data set.

The second principal component explains about 13.5% of the variance in the data set from a pattern between '*Age*' and '*Additional_charges*', with loadings of 0.7 each as well. This suggests a correlation between patient age and the additional costs variable for the patient admissions provided in the data set.

The third principal component explains 9% of the variance in the data set from a pattern between '*Lat*' and '*Population*' with loadings of -0.71 and 0.62, respectively. This suggests a correlation between geographic and demographic dimensions.

Lastly, the fourth principal component explains 8% of the variance in the data set from a pattern between '*Lng*' and '*VitD_levels*' with loadings of 0.47 and 0.52, respectively. Again suggesting a correlation between geographic and demographic dimensions.

D4: Total Variance Captured by Components

Identify the total variance captured by the four principal components.

The four principal components together make up approximately 45.5% or of the total variance in the data set. Although this is not the majority, considering the high dimensionality of the data set, capturing nearly 50% (rounded up) of the variance in just four components is important. This variance is significant considering the spread of the remaining variance is across many other components. The variables making up nearly 50% are important for finding patterns related to hospital efficiency and efficacy since they make up such significant variance in the data despite being only 8 of the 50 variables in the entire data set.

D5: Summary of Data Analysis

Summarize the results of the data analysis.

This Principal Component Analysis of the '*medical_clean*' data set identified four key principal components that explains significant patterns of the data while focusing on hospital efficiency and operations. The findings can help hospital admins and professionals optimize resource allocation and improve patient care by addressing the factors contributing to each principal component. The analysis evaluated thirteen continuous variables that explain almost 50% of the variance in the data set, consolidated to four principal components which helps with further analysis of this specific data set.

Attachments

E: Sources for Third-Party Code

DataCamp Course Resource.

F: Sources

DataCamp Course Resouce.

Vadapalli, P. (2020, November 11). PCA in Machine Learning: Assumptions, steps to apply & applications. upGrad blog. https://www.upgrad.com/blog/pca-in-machine-learning/

Whitfield, B. (2024, February 23). A step-by-step explanation of principal component analysis (PCA). Built In. https://builtin.com/data-science/step-step-explanation-principal-component-analysis

Wikimedia Foundation. (2024, May 16). Principal component analysis. Wikipedia. https://en.wikipedia.org/wiki/Principal_component_analysis

In [ ]: